METHOD AND APPARATUS FOR INITIALIZING MULTIPLE PROCESSORS RESIDING IN AN INTEGRATED CIRCUIT

DESCRIPTION

[Para 1] Field of the Present Invention

[Para 2] The present invention generally relates to integrated circuits, and more specifically, to integrated circuits having multiple processors that are to be initialized during the powering on of the integrated circuit.

[Para 3] Description of Related Art

[Para 4] Consumers continuously challenge the electronic industry to produce increasingly smaller devices while introducing even greater functionality. It has become an expectation of the consumer that these devices will become instantly available upon turning the power on.

[Para 5] These devices typically contain a myriad of integrated circuits each supporting numerous functional capabilities with the assistance of one or more processors. The initialization of these processors has been accomplished using several different solutions. The previous solutions were developed without the time constraints presented by the current instantaneous expectations of the consumer.

[Para 6] It would, therefore, be a distinct advantage to have a method and system that would initialize multiple processors residing in an integrated circuit in a time and code efficient manner. The present invention provides such a method and apparatus.

[Para 7] SUMMARY OF THE PRESENT INVENTNION

[Para 8] The present invention initializes multiple processors in an integrated circuit. Boot code is written for initialization of the processors so that there is a general/common section and specific sections for processor specific code. Each one of the processors is uniquely identifiable (in the preferred embodiment of the present invention, the strapping of a unique value to a register is used). During execution of the general section of boot code, the identity of the processor is used to execute any specific code for the identified processor in the specific section of boot code.

[Para 9] BRIEF DESCRIPTION OF THE DRAWINGS

[Para 10] The present invention will become better understood and its numerous advantages will become more apparent to those skilled in the relevant art by reference to the following drawings, in conjunction with the accompanying specification, in which:

[Para 11] Figure 1 is a block diagram illustrating an integrated circuit 102 having multiple processors P1-Pn that are initialized according to the teachings of the present invention; and

[Para 12] Figure 2, a flow chart is shown illustrating the method for initializing the processors P1-Pn of Figure 1 according to the teachings of the present invention.

[Para 13] DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT OF THE PRESENT INVENTION

[Para 14] The present invention initializes multiple processors, residing in an integrated circuit, by providing each processor with a unique identification value. This unique identification value is then used by the boot code to identify specific code to initialize the identified processor. The

incorporation of the identity of the processor into the boot code allows each of the processors to contain the same address for the boot code (i.e. starting point), and to reuse common portions of the boot code as explained in greater detail below in connection with the corresponding figures.

[Para 15] Reference now being made to Figure 1, a block diagram is shown illustrating an integrated circuit 102 having multiple processors P1-Pn that are initialized according to the teachings of the present invention. Integrated circuit 102 includes multiple processors P1 - Pn, a shared cache 110, memory 112, processor bus 114, and system bus 116.

[Para 16] The processors P1-Pn can be, for example, the PowerPC 405 processor from International Business Machines or any other processor capable of being incorporated into an integrated circuit. In the preferred embodiment of the present invention, each of the processors P1-Pn include a register that is strapped or otherwise provided with a value to uniquely identify the processor P1-Pn. Each one of the processors P1-Pn is provided with the capability of transferring and receiving data via processor bus 114.

[Para 17] Shared Cache 110 is a standard shared cache implementation having sufficient size and speed to accommodate the aspects of the present invention as explained in greater detail with the other elements of integrated circuit 102. It should be noted, however, that shared cache 110 is optional and not a necessary requirement for the present invention, but merely enhances the speed and operation of the present invention where such speed and operation are desirable (e.g. large number of processors). Shared Cache 110 is coupled to the processor bus 114 and the system bus 116.

[Para 18] Shared memory 112 is a typical memory having a sufficient size to store the boot code for the processors P1-Pn, and is coupled to system bus 116. The boot code should be located at an address that is consistent with the address provided to the processors P1-Pn for initialization. A pseudo code example of the boot code is provided in Table 1 below.

[Para 19] TABLE 1

// General Initialization Code//

perform code routines here that are common for all processors.

Read unique identification value of processor

Jump/branch to processor specific code as identified by unique identifier

//Processor Specific Code //

Processor specific code resides here

[Para 20] Reference now being made to Figure 2, a flow chart is shown illustrating the method for initializing the processors P1-Pn of Figure 1 according to the teachings of the present invention. The method begins (step 200) by providing each one the processors P1-Pn with a unique identification (step 202). In the preferred embodiment of the present invention, the unique identification is provided by strapping a register, such as a DCR register. The method continues by executing a particular start sequence for the processors P1-Pn (step 204) where each of the processors P1-Pn executes the boot code residing either in memory 112 or shared cache 110 (if retrieved by a previous processor P1-Pn) (step 204). The unique identification is used in the boot code to initialize the processor P1-Pn with any processor P1-Pn specific code.